IBM Tivoli Identity Manager Version 4.6 for z/OS Performance Tuning Guide

Issue Date:
2007 March 02 – First Edition

Publication Number:
SC23-6536-00

Table of contents

About this guide

This guide identifies some ways to tune your IBM® Tivoli Identity Manager™ system to improve performance.

Who should use this guide

Use this guide if you are responsible for installing or maintaining an IBM Tivoli Identity Manager system on z/OS. The following competencies are recommended:

- Familiarity with basic database and directory design principles.
- General knowledge of the z/OS environment.
- Understanding of how to configure and administer your directory and database servers. You may need to have your local database administrator or directory administrator perform these tunings for you.

1 Introduction

The IBM Tivoli Identity Manager product addresses the complex problem of identity management. Due to the complexity of the problem, it can be challenging to optimize the use of resources by IBM Tivoli Identity Manager – that is, to tune. This tuning guide provides a system administrator with the information needed to tune the application for your environment. Other individuals (such as IBM DB2 or the LDAP Server administrators) in your organization might offer differing advice. In our experience, your system administrators know your environment better, and their advice may be more accurate for your environment than this tuning document.

The IBM Tivoli Identity Manager product can be divided into four major components: IBM WebSphere Application Server, the IBM Tivoli Identity Manager application, IBM DB2, and IBM LDAP Server. We will address each of these separately in this document.

The IBM Tivoli Identity Manager server can be installed as either a single server or as clustered servers. A clustered environment can be considered a group of single servers with regard to tuning.

This document is a working document. As more information is gathered settings may be added, removed or changed in future editions. It is recommended that you check the IBM Web site for the most recent version. To find the most recent version, go to http://www.ibm.com/support/us. Type “ITIM Tuning Guide” in the search box under **Search technical support**, and click **Search**.

1.1 Vital tunings

There are several thousand different parameters that you can modify to tune WebSphere Application Server, the IBM Tivoli Identity Manager product, directory servers, and database servers. This tuning guide discusses a small subset of these parameters that have proven effective during performance testing.

If you are setting up an acceptance or production environment, read each section and perform the applicable tunings for your systems. If you are setting up a test environment and want to get started as quickly as possible, focus on these areas:

- IBM DB2 - Buffer pools
- IBM DB2 - Reorg and Runstats
  Note: The database statistics tunings are a vital part of the IBM Tivoli Identity Manager product performance.
- IBM LDAP Server – Indexing

1.2 Initial tunings

Most of these tunings can be implemented in a newly deployed environment or an environment that is already deployed.

It is recommended that you execute `runstats` each time you add significant numbers of users to your databases. Failure to keep your database statistics up to date can cause IBM DB2 to use non-optimal paths when accessing data. See the IBM DB2 - Reorg and Runstats section for more information.

1.3 Resource allocation

Tuning values are more complex to manage when more than one middleware component is running on a given system; for example, having the IBM Tivoli Identity Manager server, IBM DB2, and IBM LDAP Server all on the same server. Regardless of configuration, it is important to calibrate the following resources so that they are not over-allocated.

1.3.1 Memory

All middleware components allow you to adjust how much memory they will use. When calculating how to allocate memory to middleware components, keep these considerations in mind:

- Configuring middleware memory settings too high such that the total configured value exceeds available physical memory can result in the operating system swapping memory out to disk. **This will result in extremely poor performance and should be avoided.** After setting up or changing the memory values for the middleware, monitor the memory and swap space used to ensure that nothing is being swapped out to disk. If it is, adjust your memory settings to correct.
- A large part of the WebSphere Application Server's memory usage is the JVM size. However, the size of the JVM does not set an upper bound on the amount of memory that the WebSphere Application Server may use. See the IBM WebSphere Application Server section.

1.3.2 CPU

All the components of the IBM Tivoli Identity Manager product (IBM Tivoli Identity Manager application, WebSphere Application Server, database server, and directory server) are CPU-intensive. Normally, batch processes such as DSML feeds are less CPU intensive than interactive commands such as changing passwords. Operations involving workflow, such as account creation, are very computationally intensive, especially when customized workflow processes are enabled. zAAP processors, if available, should be utilized in the z/OS instances supporting IBM Tivoli Identity Manager.

1.3.3 Disk space

Each of the middleware components uses different amounts of disk space for various purposes.

- WebSphere Application Server and the IBM Tivoli Identity Manager application use disk space beyond their installation size because of log files (such as the `msg.log` and `trace.log` files) and WebSphere MQ queues. Adjust the number of archives and size of the `msg.log` and `trace.log` files in the `enRoleLogging.properties` file. Make sure that WebSphere MQ has enough disk space for its processing logs (not error logs) to grow. The IBM Tivoli Identity Manager server pushes many entries onto the queues during large provisioning changes, causing the queues to grow.
- IBM DB2 archive logs can consume a great deal of space for large transactions. For example, automatically provisioning an IBM Tivoli Identity Manager account for 50k people resulted in 13.5 GB of space being used. Only 2.7 GB was for account storage (both inside the LDAP Server and the historical logging in IBM DB2), the remainder, roughly 80%, was used by IBM DB2 archive logs. Frequent purging of IBM DB2 archive logs may be required for busy systems.

2 IBM WebSphere Application Server

Regardless of the installation type (single server or cluster), the IBM Tivoli Identity Manager server can be thought of as two components: WebSphere Application Server (the J2EE application server running the application) and the IBM Tivoli Identity Manager application itself. Both components need to be tuned.

WebSphere Application Server allows you to use a variety of settings to tune your environment. This document discusses the timeouts and Java Messaging Service (JMS) queue endpoints.

2.1 Java virtual machine (JVM) size

By default, WebSphere Application Server sets the maximum JVM size to 256 MB. This value is too small for the IBM Tivoli Identity Manager product to run beyond a basic concept test and should be increased to a minimum of 768 MB. If your server has adequate available RAM increase this value to as much as 1.5 GB. For large reconciliations or role and policy evaluations, the default values will not be enough memory to complete these tasks.

The maximum JVM size is not the actual maximum allocated size of the Java heap – as much as 15% is allocated to a portion of the heap for the system's use. IBM recommends that you not use a value higher than 1.5 GB even if your system has the available memory.

Do not set the JVM heap size to be larger than the physical RAM. The WebSphere Application Server suffers significant performance degradation if the operating system swaps out the JVM to swap space. Consequences of this include very slow user interface (UI) performance, transaction roll backs, timeouts, and high disk utilization.

Determining the values

initial_jvm_heap_size – The initial size of the JVM heap in megabytes. Recommended value: 256 MB.

max_jvm_heap_size – The maximum size of the JVM heap in megabytes. Recommended value: 768 MB.

Setting the values

1) Open the Administration Console.
2) Expand the **Servers** list in the navigation pane.
3) Select **Application Servers** in the navigation pane.
4) Select the server to manage.
5) Select **Process Definition** from the **Additional Properties** pane at the bottom.
6) Select **Java Virtual Machine** from the **Additional Properties** pane at the bottom.
7) Set the **Initial Heap Size** to *initial_jvm_heap_size*.
8) Set the **Maximum Heap Size** to *max_jvm_heap_size*.
9) Repeat this procedure for each IBM Tivoli Identity Manager server.

Stop and restart each Application Server for these changes to take effect.

2.2 Workload management (WLM) timeout

The WebSphere Application Server on z/OS provides a timeout value for how long it should wait for IIOP requests to complete. WebSphere Application Server uses the workload management (WLM) timeout value to terminate hung threads thereby preventing the hung thread from holding onto resources needed

by other processes. When the WLM timeout is reached the thread is killed resulting in the process being killed.

IBM Tivoli Identity Manager can initiate long-running IIOP requests during processes like reconciliations. To allow long-running reconciliations to complete we recommend disabling the WLM timeout.

Determining the values

wlm_timeout – The time in seconds that IIOP requests wait on the queue before being terminated. Default value: 300. Recommended value: 0 (disabled).

Setting the values

1) Open the Administration Console.
2) Expand the **Servers** list in the navigation pane.
3) Select **Application Servers** in the navigation pane.
4) Select the server to manage.
5) Select **Container Services**.
6) Select **ORB Service**.
7) Select **z/OS additional** settings.
8) Change **WLM timeout** to *wlm_timeout*.
9) Repeat this procedure for each IBM Tivoli Identity Manager server.

Stop and restart each Application Server for these changes to take effect.

2.3 Message driven bean (MDB) request timeout

The message driven bean (MDB) request timeout controls how much time MDBs should be limited to. Policy evaluations can often run over the default timeout and should be increased.

Determining the values

mdb_request_timeout – The time in seconds allotted to MDB requests. Default value: 1200. Recommended value: 7200.

Setting the values

1) Open the Administration Console.
2) Expand the **Servers** list in the navigation pane.
3) Select **Application Servers** in the navigation pane.
4) Select the server to manage.
5) Select **Custom Properties**.
6) Change the value of **control_region_mdb_request_timeout** to *mdb_timeout*.
7) Repeat this procedure for each IBM Tivoli Identity Manager server.

Stop and restart each Application Server for these changes to take effect.

2.4 Transaction timeout

Complex or long-running transactions within IBM Tivoli Identity Manager can often run past the default transaction timeout. It is recommend that the default transaction timeout be increased.

Determining the values

transaction_timeout – The time in seconds before a transaction timeouts. Default value: 300. Recommended value: 12000.

Setting the values

1) Open the Administration Console.
2) Expand the **Servers** list in the navigation pane.
3) Select **Application Servers** in the navigation pane.
4) Select the server to manage.
5) Select **Container Services**.
6) Select **Transaction Services**.
7) Change **Maximum Transaction Timeout** to *transaction_timeout*.
8) Repeat this procedure for each IBM Tivoli Identity Manager server.

Stop and restart each Application Server for these changes to take effect.

3 IBM Tivoli Identity Manager application

The IBM Tivoli Identity Manager application includes several configuration files that provide an area for tuning various parts of the application's performance. These are in the `data/` directory under the IBM Tivoli Identity Manager product home directory.

3.1 Recycle bin

When objects such as people, accounts, roles, and provisioning policies are deleted from the IBM Tivoli Identity Manager system using either the graphical user interface (GUI) or the application program interface (API), these objects are not removed from the underlying directory server but rather moved into the recycle bin. The recycle bin is implemented as the following LDAP container:

```
ou=recycleBin, ou=itim, ou=<tenant>, <suffix>
```

When LDAP entries are moved under this DN due to a deletion, the attribute `erIsDeleted` is set to the value `Y` to enable IBM Tivoli Identity Manager to identify these objects as entries it should neither display to the user nor act upon. Because of the LDAP search filter that IBM Tivoli Identity Manager uses, having a large number of entries in the recycle bin can negatively impact performance. It is recommended that the size of the recycle bin be kept as small as possible for optimum performance.

There are several ways to remove entries from the recycle bin. IBM Tivoli Identity Manager includes a script that will delete entries in the recycle bin older than a specified age range. See the discussion of the recycle bin age limit in *IBM Tivoli Identity Manager Server Installation and Configuration Guide for WebSphere Environments* for more information.

An alternate method is to use an LDAP display tool to view the entries and delete them directly in the directory server. Be careful to only delete the deleted entries themselves and not the `ou=recycleBin` container. Similarly, it is possible to use a combination of the ldapsearch and ldapdelete commands to delete entries. For example:

```
ldapsearch -h <host> -p <port> -D <user> -w <password> \
   -b "ou=recycleBin,ou=itim,ou=<tenant>,<suffix>" -s sub "erisdeleted=Y" dn | \
ldapdelete -h <host> -p <port> -D <user> -w <password>
```

After deleting entries from the recycle bin, run `runstats` to make IBM DB2 pick up the changes. See the IBM LDAP Server – Runstats section for more information.

3.2 Reconciliations

Reconciliations are resource-intensive operations and can take a while for services with a large account population. Limiting the number of attributes returned by the adapter and processed by IBM Tivoli Identity Manager can improve reconciliation performance. Large reconciliations may also exceed the default Max Duration and if so the value can be increased.

3.2.1 Threads

When processing DSML feeds, IBM Tivoli Identity Manager creates threads to process the data. The number of threads may need to be adjusted to optimize performance because of the widely varying workload that differently defined reconciliation jobs exhibit.

Determining the values

num_recon_threads – The number of threads used when processing DSML feeds.
Recommended value: 2 for DSML feeds with workflow, 3 for DSML feeds without workflow.

Setting the values

Edit the `enRole.properties` file and update the value of `enrole.reconciliation.threadcount` to *num_recon_threads*.

3.2.2 Limiting attributes returned from the adapter

Some adapters (such as the adapter for Microsoft Active Directory) can limit the attributes that are returned to the IBM Tivoli Identity Manager server during reconciliations. This can reduce the amount of work required by the adapter to retrieve or calculate the values of the attributes as well as amount of data sent back to IBM Tivoli Identity Manager.

Consult the adapter documentation for information specific to that adapter. See also the IBM Tivoli Identity Manager adapters section.

3.2.3 Limiting the attributes evaluated

During reconciliation, any attributes that were identified as having been changed are updated within the IBM Tivoli Identity Manager directory server. Before this update takes place, the new value is evaluated against the provisioning policy that governs the account to ensure that the change is allowed within the policy, and if not, a policy enforcement is triggered. Any change to the account will trigger the policy evaluation for that account regardless if the change would invalidate the policy.

To reduce the number of policy evaluations, limit the attributes that are evaluated during reconciliation. Some endpoints (such as Microsoft Active Directory) contain attributes that change frequently but are seldom used to enforce policy, such as last logon time. If these attributes are required, consider setting up a second reconciliation to reconcile these attributes on a more infrequent schedule and remove them from the more frequently running reconciliations. If possible, reconcile only those attributes that are required for policy evaluation.

Determining the values

> *excluded_attributes* – The list of attributes that are returned from the adapter to exclude from processing within IBM Tivoli Identity Manager. Ideally these would be all attributes except those that are required for policy evaluation.

Setting the values

1) Log into IBM Tivoli Identity Manager as a user with sufficient privileges to edit the service.
2) Select the **Provisioning** tab.
3) Navigate to the service in the organizational tree.
4) Select the service to modify.
5) Select **Reconciliation**.
6) Select the reconciliation schedule to modify.
7) Select the **Query** tab.
8) Move all *excluded_attributes* from to the Excluded Attributes list.
9) Click **Submit**.

3.2.4 Maximum duration

Large reconciliations can sometimes exceed the default maximum duration as specified in the reconciliation schedule. When this limit is reached, the reconciliation is aborted. Increase the limit to allow longer-running reconciliations to complete.

Determining the values

max_duration – The maximum time in minutes that the reconciliation should run. To calculate this value, do an initial run with a very large duration and measure the time required. Consider setting the maximum duration to 10% above this time. Default value: 600.

Setting the values

1) Log into IBM Tivoli Identity Manager as a user with sufficient privileges to edit the service.
2) Select the **Provisioning** tab.
3) Navigate to the service in the organizational tree.
4) Select the service to modify.
5) Select **Reconciliation**.
6) Select the reconciliation schedule to modify.
7) Set the **Max Duration** to *max_duration*.
8) Click **Submit**.

4 IBM Tivoli Identity Manager adapters

It is sometimes necessary to tune the IBM Tivoli Identity Manager adapters when doing large provisioning changes or reconciliations. This section should supplement, not supersede, the documentation included with the adapter.

4.1 Microsoft Active Directory

The Microsoft Active Directory adapter returns attributes to IBM Tivoli Identity Manager that are not directly retrieved from Active Directory, but rather calculated from other Windows sources. Querying these external sources can slow down Active Directory reconciliations and can be disabled if these attributes are not needed.

The Home Directory Security and Mailbox Permissions are two such attributes. Retrieving this information requires looking up the appropriate access control entry, which is a costly operation. Setting `ReconHomeDirSecurity` and `ReconMailboxPermissions` to FALSE in the adapter registry will disable this overhead.

Working with Windows Terminal Services (WTS) attributes can slow down provisioning and reconciliation as well. There are two adapter registry keys that control access to these attributes:

- `WtsEnabled` – This key controls the adapters' access to WTS attributes. If this key is enabled (set to TRUE) the adapter will have access to provision and reconcile WTS attributes. If this key is disabled (set to FALSE) the adapter will not provision WTS attributes if requested, nor will it return them during reconciliation. The default value for this key is FALSE.

- `WtsDisableSearch` – This key controls whether the adapter will return WTS attributes during a reconciliation (a "search" from the adapter's perspective). If this key is enabled (set to TRUE), WTS attributes will not be returned in a reconciliation but the attributes will still be updated in account provisions. If this key is disabled (set to FALSE), WTS attributes will be returned in a reconciliation. This key only applies if the `WtsEnabled` key is set to TRUE. The default value for this key is TRUE.

5 IBM DB2

Tuning IBM DB2 to run with the IBM Tivoli Identity Manager product involves adjusting the buffer pools, modifying the number of connections, modifying internal database values, adding table space, adjusting logs, indexing, and running runstats.

The tuning JCL provided applies to the z/OS 1.6 LDAP server. The z/OS 1.8 LDAP server was not available at the time this document was prepared.

5.1 APARs

Two ODBC APARs have been linked to poor LDAP performance on z/OS. It is recommended that these APARs be installed on the system:

- PK21695 – Mutex lock contention
- PK17652 – High CLI CPU utilization

5.2 Buffer pools

Tuning the IBM DB2 buffer pools has shown to decrease deadlocks and improve overall performance. IBM Tivoli Identity Manager includes three JCL scripts to tune the IBM DB2 buffer pools based on which database is being tuned and how WebSphere Application Server is set up.

Determining the values

#	JCL location	Description
1	hlg.SAERSAMP(AERTNDBC)	Tune ITIM database when using a WAS cluster
2	hlg.SAERSAMP(AERTNDBS)	Tune ITIM database when using a single WAS server
3	hlg.SAERSAMP(AERTNLDP)	Tune LDAP database

Identify which JCL script is needed to tune your Tivoli Identity Manager database (either 1 or 2). JCL script 3 is used for the LDAP database for both single and clustered WebSphere configurations.

Setting the values

1) Edit the JCL to conform to your installation's standards.
2) Update DB2 schema names, such as the database name, to match your environment.
3) Submit the JCL for execution.

5.3 Idle thread timeout

DB2 continually checks for idle threads to cancel. Sometimes during SQL queries that return large amounts of data, if there is a delay between the sending of one DRDA query block and another, an abend 04E reason code 00D3003B, may occur, indicating that the idle thread timeout was exceeded. Increase the idle thread timeout to avoid this error.

Determining the values

idle_thread_timeout – Total time in seconds before an idle thread will timeout. Default value: 120. Recommended value: 7200. This value is in the DB2 `DSNTIJUZ` customization job, located on the `IDTHTOIN` operand for the `DSN6FAC` macro

Setting the values

1) Copy `DSN.V8R1M0.NEW.SDSNSAMP(#02TIJUZ)` to another dataset and/or member for updating.
2) Locate the line containing `DSNTIZP` and delete everything to the end of the file, including this line.
3) Locate the `IDTHTOIN` operand for the `DSN6FAC` macro.
4) Change the value to *idle_thread_timeout* as determined above, with care not to delete the continuations in column 72.
5) Locate the line containing `SYSLMOD`.
6) Update the DSN value on the following line to a data set where you have modification permissions (for example: `DSN=DSN81.SDSNEXIT`).
7) Submit the job and ensure it runs with a condition code zero.

Restart IBM DB2 subsystem for the change to take effect.

5.4 Locks per user limit

During policy analysis, DB2 error 00C90096 may be seen in the `trace.log` file. This occurs when IBM Tivoli Identity Manager attempts to remove a large number of rows from a temporary table and the number of rows exceed the locks per user limit. This can be avoided by increasing the locks per user limit.

Determining the values

locks_per_user_limit – Total number of locks a user is allowed to hold at one time. Default value: 10000. Recommended value: 100000 or higher.

Setting the values

If implementing IBM DB2 for the first time, or creating a separate instance, this parameter can be increased by editing the **LOCKS PER USER** value.

To increase the limit for an existing installation:

1) Edit the existing `DSNTIJUZ` job.
2) Update `NUMLKUS` in the `DSN6SPRM` macro using the procedure described in the Idle thread timeout section.
3) Submit the job to rebuild the DB2 startup parameters.

Restart IBM DB2 subsystem for the change to take effect.

5.5 Active log duplexing

Disabling the IBM DB2 active log duplexing has shown to increase account creation throughput by up to 62%. Changing this setting depends upon your installation's procedures and conventions.

Determining the values

active_log_duplex – Enable active logging duplexing? Default value: YES. Recommended value: NO.

Setting the values

1) Edit the existing `DSNTIJUZ` job.
2) Update `TWOACTV` to *active_log_duplex* in the `DSN6LOGP` macro using the procedure described in the Idle thread timeout section.
3) Submit the job to rebuild the DB2 startup parameters.

Restart IBM DB2 subsystem for the change to take effect.

5.6 Reorg and Runstats

Statistics on the number of rows in the tables and what indexes are available are required for IBM DB2 to efficiently fulfill queries. It is important to update these table and index statistics after large Directory Server Markup Language (DSML) loads, HR feeds, and reconciliations.

IBM Tivoli Identity manager ships with five JCL to execute `reorg` and `runstats` against the IBM DB2 databases.

In addition to running `runstats` on all tables within the database, we also manually update the statistics table for the ACTIVITY, PROCESS, PROCESSDATA, and SCHEDULED_MESSAGE tables to ensure a minimum cardinality. Setting a minimum cardinality on these tables helps the IBM DB2 query optimizer and can decrease locking issues in the database.

Determining the values

JCL location	Description
hlg.SAERSAMP(AERROITM)	REORG for ITIM database
hlg.SAERSAMP(AERROLDP)	REORG for LDAP database
hlg.SAERSAMP(AERROLDC)	REORG for LDAP changelog database
hlg.SAERSAMP(AERRSITM)	RUNSTATS for ITIM database
hlg.SAERSAMP(AERRSLDP)	RUNSTATS for LDAP database

The REORG scripts above will reorganize both table spaces and indexes, improving data access performance and reclaiming fragmented space. In addition, the REORG scripts will execute a RUNSTAT for each table space after the REORG.

Caution: The REORG scripts should only be executed when the table spaces are not in use by stopping the WebSphere Application Server and LDAP first. Failure to do this will result in the table spaces in an intermediate state and will require additional attention.

Use RUNSTATS scripts on idle or lightly-used databases because it requires update locking on the system statistics table to update the database statistics. A database with a heavy load might experience transaction roll backs due to the system acquiring locks on the tables that are used by the database optimizer to fulfill queries.

Note: The LDAP REORG and RUNSTATS JCL scripts rely on the DIR_DESCX2 index. See the Indexing section for more information.

Setting the values

1) Edit the JCL to conform to your installation's standards.
2) Update DB2 schema names, such as the database name, to match your environment.
3) Submit the JCL for execution.
4) Confirm the job's completion code for success.

After running RUNSTATS on the ITIM database, the following should be run against the ITIM database to update the statistics for the workflow tables. The following SQL statements require DB2 system administrator privileges to perform.

```
UPDATE SYSIBM.SYSTABLES SET CARD = 50000
   WHERE CARD < 50000 AND CREATOR = 'ENROLE' AND NAME = 'ACTIVITY';
UPDATE SYSIBM.SYSTABLES SET CARD = 50000
   WHERE CARD < 50000 AND CREATOR = 'ENROLE' AND NAME = 'PROCESS';
UPDATE SYSIBM.SYSTABLES SET CARD = 50000
   WHERE CARD < 50000 AND CREATOR = 'ENROLE' AND NAME = 'PROCESSDATA';
UPDATE SYSIBM.SYSTABLES SET CARD = 50000
   WHERE CARD < 50000 AND CREATOR = 'ENROLE' AND NAME = 'SCHEDULED_MESSAGE';
```

5.7 Additional ZPARMS

The WebSphere Application Server 6.0 Developer's Guide recommends updating the following ZPARMS in addition to the ones mentioned elsewhere in this document. For more information on these changes, please see the WebSphere Application Server 6.0 Developer's Guide.

ZPARM	Default value	Recommended value
MAXKEEPD	5000	16000
CHKFREQ	500000	16000000
CONDBAT	10000	400
CTHREAD	200	1200
DBPROTCL	DRDA	PRIVATE
IDBACK	50	1800
STORTIME	180	600

6 IBM LDAP Server

The IBM LDAP Server is a component of the Integrated Security Services base element in z/OS R 1.6 and 1.7, not to be confused with the IBM Tivoli Directory Server released with z/OS 1.8.

6.1 APARs

The following APARs are recommended to fix insert and update failures when using IBM Tivoli Identity Manager:

- OA14765 – Addresses LDAP deadlocks
- OA17432 – Moves DIR_MISC table to MISCTS tables pace

6.2 Cache sizes

The LDAP Server has internal caches to allow quick access to frequently accessed entries in memory rather than accessing the values from the disk. Better performance can be obtained by increasing the size of the caches.

The LDAP Server allows you to control how many entries the entry cache can store but does not restrict the size of the cache. The size of each entry in the cache is based on the number and the size of attributes that a given LDAP entry has. Typically, many entries are users and their accounts, which have a fairly constant size. When setting the value for the entry cache, calculate the size of the average entry and divide that into the amount of memory used by the LDAP Server process. Users with few attributes can generate entry sizes that are approximately 4 KB where users with more attributes can generate entry sizes around 9k.

Determining the values

dn_cache_size – Size of the DN cache. Default value: 1000. Recommended value: 75000.

dn_to_eid_cache_size – Size of the DN to EID cache. Default value: 1000. Recommended value: 75000.

entry_cache_size – Size of the entry cache. Default value: 1000. Recommended value: 75000.

Note: The recommended values above were determined by assuming 15000 users each with 5 accounts for a total of 75000 objects. You may need to increase this value for larger populations.

Setting the values

1) Edit `GLD.CNFOUT(SLAPDCNF)`
2) Modify the **dnCacheSize** value to *dn_cache_size*.
3) Modify the **dnToEidCacheSize** value to *dn_to_eid_cache_size*.
4) Modify the **entryCacheSize** value to *entry_cache_size*.
5) Restart LDAPSRV

6.3 Max connections

To ensure that IBM Tivoli Identity Manager can connect to the directory server using all available connections, ensure the maximum number of LDAP connections is greater than the size of the LDAP connection pool for Tivoli Identity Manager.

Determining the values

max_connections – The maximum number of connections that the LDAP server will accept. Set this value to 20 more than the `enrole.connectionpool.maxpoolsize` specified in the `enRole.properties` file.

Setting the values

1) Edit `GLD.CNFOUT(SLAPDCNF)`
2) Modify the **maxConnections** value to *max_connections*.
3) Restart LDAPSRV

6.4 Changelog limits

The LDAP Server changelog can be limited either by the number of entries or the maximum age of an entry. High LDAP add or modify operation rates initiated by Tivoli Identity Manager can result in lock escalations due to the large volume of entries being added and removed from the changelog. For this reason, it is recommended that the changelog be limited by the maximum age of an entry (changeLogMaxAge) instead of the number of entries (changeLogMaxEntries). Both of these values can be set in `GLD.CNFOUT(SLAPDCNF)`.

6.5 Row locking on SEARCHTS

To improve locking parallelism, particularly on single server installations, it is recommended to change the locking on SEARCHTS table space in GLDDB and GLDDBG databases to row level. This can be done with the following DB2 commands:

```
ALTER TABLESPACE GLDDB.SEARCHTS LOCKSIZE ROW;
ALTER TABLESPACE GLDDBG.SEARCHTS LOCKSIZE ROW;
```

These commands can be executed from SPUFI interface, option 6 using SYSADM login.

6.6 Indexing

Indexing the attributes that applications search on increases LDAP Server performance. The LDAP Server indexes are automatically translated into IBM DB2 indexes when you update the LDAP Server schema for those attributes.

If you extend the LDAP schema in LDAP Server to include additional attributes, index those attributes that you will search for. Any filter in the Tivoli Identity Manager application (such as with dynamic roles) is translated into a search string for the LDAP Server.

The Tivoli Identity Manager application frequently searches against the organization (o), organizational unit (ou), and owner attributes.

After updating the LDAP schema, run DB2 `runstats` on the database to update the statistics for the newly created indexes.

In addition, the following DB2 index has shown to increase performance and is required by the LDAP JCL REORG and RUNSTATS scripts:

```
CREATE UNIQUE INDEX LDAPSRV.DIR_DESCX2
    ON LDAPSRV.DIR_DESC( AEID, DEID )
    USING STOGROUP SYSDEFLT PRIQTY 22000 SECQTY 10000
    CLOSE NO  BUFFERPOOL BP1 DEFER NO;
```

6.7 Runstats

See the IBM DB2 - Reorg and Runstats section.

7 Best practices

The IBM Tivoli Identity Manager product can be set up and configured in many ways. The following are some suggested best practices to help guide you in setting up your environment.

- Each agent modifies the LDAP schema by adding new attributes to support a new service. These attributes are created without indexes, and for services that service thousand of users, a large benefit can be achieved by adding indexes to attributes with many members.
- Complicated provisioning policies can result in complicated directory and database queries with poor performance. Policies with small numbers of roles and services will perform best.
- Dynamic roles affect people in a given scope, either one-level or subtree. When a person object within that scope is modified or added, that role must be re-evaluated. This is true for every dynamic role in the system. For instance, if there are three dynamic roles with subtree scope and a person object within that scope is updated, all three dynamic roles will be re-evaluated. For this reason, it is recommended that you limit the number of dynamic roles, either by number or by scope, that affect people that are modified frequently. It doesn't matter if the dynamic role ends up enrolling the person or not, the evaluation itself is the performance-impacting overhead.
- Limiting the scope (via placement within the organizational tree) and number of ACIs will increase performance by requiring fewer evaluations. When doing a person search via the APIs, be sure to limit the scope of your search to be as narrow as possible to avoid the system evaluating more ACIs than necessary.
- When enabling WebSphere global security, do not enable Java 2 security unless it is required for your environment. Enabling WebSphere global security automatically enables Java 2 security unless it is explicitly disabled. Having Java 2 security enabled can cause a significant performance degradation to IBM Tivoli Identity Manager.

8 Regular maintenance

To maintain optimal performance for your IBM Tivoli Identity Manager environment, regular maintenance is required.

- Regularly empty the IBM Tivoli Identity Manager recycle bin. As the number of objects in the recycle bin increase LDAP performance can degrade. The frequency at which the recycle bin is emptied depends on how frequently deletes occur in the system. Ideally, the recycle bin would contain fewer than 100 items. See IBM Tivoli Identity Manager application – Recycle bin for more information on how to empty the recycle bin.
- Update database statistics after a large number of updates. Updating database statistics in the underlying databases can significantly improve performance and should be done on a weekly basis for most environments. This applies to the IBM DB2 database if using LDAP Server as well as the underlying IBM DB2 of the IBM Tivoli Identity Manager application. See the appropriate database sections on how to update database statistics.

9 Other resources

You will find the following resources useful for further tuning of IBM Tivoli Identity Manager:

IBM Tivoli Identity Manager 4.6 Performance Tuning Guide for distributed:

http://publib.boulder.ibm.com/tividd/td/ITIM/SC23-5272-02/en_US/PDF/SC23-5272-02.pdf

z/OS Integrated Security Services LDAP Server Administration and Use

http://publibz.boulder.ibm.com/cgi-bin/bookmgr_OS390/BOOKS/GLDA2A31/2.17?SHELF=ICHZBK60&DT=20050118133745

DB2 Universal Database for z/OS Version 8 Administration Guide

http://publib.boulder.ibm.com/cgi-bin/bookmgr/BOOKS/dsnagj12/5.0?DT=20050329172544

WebSphere Application Server for z/OS Version 6.0.2 Tuning Guide

ftp://ftp.software.ibm.com/software/webserver/appserv/library/v60/BOS5J112.pdf